# SYNTAX ZX80®

A PUBLICATION OF **THE HARVARD GROUP**

VOL.2 NO.10 ISSN 0273-2696 OCT., 1981

IN THIS ISSUE:



## SINCLAIR PRODUCT UPDATES

Sinclair's ZX81 computer reaches the US on Oct.7, 1981. After then, you can order a ZX81 for immediate delivery by calling 800/543-3000 (operator 509). Assembled ZX81s will cost $149.95; kits are $99.95.

Nigel Searle of Sinclair expects up to 5000 8K ROMs by the fourth week in September, although he promises no exact date. That quantity will fill all standing orders for new and replacement 8K ROMs. He also expects 16K RAMs to arrive around mid-October. When the units arrive, the 800 number will again take orders for them. Try calling (operator 509) to check availability.

## CAI PERIPHERALS UPDATE

CAI Instruments will ship their first Widgets the week of Sept. 21, according to Bob Swann of CAI. Because of development problems, Widget prices have changed. Widget alone--$79. With either the tape drive or printer, Widget is $69. With both other devices, Widget is $59. Printers will go out about 10 days after the Widgets, and tape drives about a month later.

## ATTENTION EARLY SUBSCRIBERS

If you received your white renewal notice before the Sept. SYNTAX with the free binder offer--fear not. We had not finalized our binder plans when the first white cards went out. All of you who return your white cards quickly will receive a binder. We'll mail them around Oct. 26.

If you haven't gotten your white renewal card in the mail yet, watch for it. Return the white card with your renewal order within 2 weeks of our postmark and we'll send you, absolutely free, a sturdy magazine binder to hold your issues. This offer does not apply to later notices, so act promptly.

SYNTAX ERROR: Ian Logan reported 2 errors we made in his article, Machine Code Print Routine (Sept. 81). The fourth paragraph should start, "Try running this 8K print...", not 4K. The thirteenth line of the machine code should read 3E00 LD A,' ', not 3E20.

## CHANGES TO SUPER ZX80 INVASION

If you have Softsync's Super ZX80 Invasion game cassette (reviewed May 81), you can add new levels of difficulty.

Load the game and type 100 (NL) to delete the REM statement. A portion of the BASIC program is now displayed.

Delete lines 120, 150, 200 and 510. Enter these lines:

```
 275 PRINT "0. EXPERT"
 300 PRINT "1. ADVANCED"
 425 PRINT "4. FIRST TIMERS"
 100 REM ENTER GO TO 1...N/L
```

Save the program on another cassette. You now have 5 different skill levels instead of the 3 supplied. You can also add other levels from -5 to 5. Numbers less than -5 or greater than 5 will result in an arithmetic overflow.

Lance M. Ward, Okemos, MI

## CHANGES TO SEPTEMBER BUDGET PROGRAM

Quentin Smith asked that we publish these changes to his ZX80 Budget program (Sept.81):

Delete line 2015 and change line 2020 to GO TO 160. Otherwise the PRINT statement in line 2010 will not stay on the screen long enough to read.

Add these lines to print the leading 0 for cents less than 10:

```
 335 IF Z(J)<10 THEN GO TO 345
 343 GO TO 350
 345 PRINT ,X(J),Y(J);".";"0";Z(
J)
 385 IF E<10 THEN GO TO 395
 393 GO TO 400
 395 PRINT "TOTAL",D;".";"0";E
New Syntactic Sum:  898, 4K
```

## CONTACTING AUTHORS

To respect your privacy, we don't publish the addresses or phone numbers of contributors. If you would like to contact the author of a letter or program in SYNTAX, drop us a line or call, giving the name and home town of the person you want to contact. We'll put him/her in touch with you. If you're an author and would like comments from others, just tell us to include your address or phone number with your letter or program.

### OUR POLICY ON CONTRIBUTED MATERIAL

SYNTAX ZX80 invites you to express opinions related to the ZX80 and the newsletter. We will print, as space allows, letters discussing items of general interest. Of course, we reserve the right to edit letters to a suitable length and to refuse publication of any material.

We welcome program listings for all levels of expertise. Programs can be for any fun or useful purpose. We will test run each one before publishing it, but we will not debug programs; please send only workable listings.

In return for your listing, we will pay you a token fee of $2.00 per program we use. This payment gives us the nonexclusive right to use that program in any form, world-wide. This means you can still use it, sell it, or give it away, and so can we.

We will consider submissions of news and hardware or software reviews. Please keep articles short (350-400 words). Again, we reserve the right to edit accepted articles to a suitable length. We will pay 7 cents per 6 characters, including spaces and punctuation, for accepted articles.

When you send in programs for possible publication in SYNTAX, please include the following information:

- How to operate the program, including what to input if it does not contain prompts.
- Whether you can run the program over again and how.
- How to exit the program.
- The Syntactic Sum (using the Syntactic Sum program in the February, 1981, issue).
- Whether it fits in 1K or 2K RAM (or 16K when available).
- Whether it uses the 4K or 8K ROM.

We pay for this explanatory text at the same rate as for articles in addition to payment for the program itself.

If you want us to return your original program listing or article, please include a self-addressed, stamped envelope. Otherwise, we cannot return submitted material.

SYNTAX ZX80 is published monthly by Syntax ZX80, Inc., a wholly-owned subsidiary of The Harvard Group, RD 2, Box 457, Bolton Road, Harvard, MA 01451. Telephone 617/456-3661. Subscriptions: 12 issues, $25. Single issue, $4.

**Publisher:** Kirtland H Olson
**Editor:** Ann L. Zevnik
**Associate Editor:** Susan G. Barber
**Technical consultant:** Eric K. Olson

Printed by Joseph E. Marhefka, Jr.
Clinton Offset Printers
Clinton, MA 01510

## TROUBLE-SHOOTING YOUR MICROACE

You've put together your MicroAce and examined the PC board solder connections--everything looks good. Power up--you get a white screen and no cursor! What do you do now?

Here is a chart compiled by removing each IC chip from a working MicroAce and observing the effect. It may provide you with a clue to the problem if your computer malfunctions.

In the kit I built, IC U18 was bad. But I also had another problem. After installing a new U18 chip, the screen flickered and incomplete entries would appear. Touching capacitor C12 caused the screen to go white. Touching the solder side of the PC board gave the same result. Capacitor C12 (47 picofarad) was bad. I installed a new 100 picofarad capacitor, which corrected the problem.

Irving Chaiet, New Paltz, NY

See list of equivalent ZX80/MicroAce components next page. Note there is not a perfect correspondence between MicroAce ICs 15&16 and ZX80 ICs 14&15, nor between MicroAce ICs 21&22 and ZX80 ICs 11 &12. These ICs' pins are connected differently, although they perform the same functions.—AZ

| CHIP REMOVED | OBSERVATIONS |
|---|---|
| U6 74LS373N | screen is white with slight vertical bars, blinks when a key is pressed. |
| U7 74LS157 | white screen, no blink response when a key is pressed, no vertical bars in picture. |
| U8 74LS157 | whistle sound in TV, streaks in picture, no blink. |
| U9 74LS157 | streaks in picture, whistling sound, no blink. |
| U10 74LS165 | white screen with vertical bars, cursor appears but with no K, keyboard responds by moving cursor across screen. |
| U11 74LS365 | normal screen display, no keyboard response. |
| U13 74LS74 | streaks in screen, no sync, no key blink response. |
| U14 74LS93 | white screen, white cursor at bottom with no K, key responds by moving cursor. |
| U15 74LS05 | streaks in screen, no vertical lines, no key blink. |
| U16 74LS05 | streaks in screen, putt-putt noise in TV sound |
| U17 74LS32 (only in 2K) | streaks in screen, no key response. |
| U18 74LS86P | white screen, no key response. |
| U19 74LS04 | no sync, tick-tick noise, no key response. |
| U20 74LS10 | vertical lines on screen, no keyboard response, no cursor, screen breaks up into streaks when key is pressed. |
| U21 74LS00 | black cursor K and vertical stripes, programs run, computer works. |
| U22 74LS00 | no sync, flashing dots on screen, vertical roll. |
| U23 74LS32 | no vertical sync, picture rolls, no keyboard response |
| U24 74LS74N | blank white screen, no keyboard response. |

CPU chip U1, ROM chip U12, and static memory chips U2-5 were not tested.

## MICROACE-ZX80 COMPONENT EQUIVALENCE

| MicroAce IC # | Part # | ZX80 IC # |
|---|---|---|
| U 1 | D780C | 1 |
| U 2 | UPD2114LC | 3 |
| U 3 | UPD2114LC | 4 |
| U 6 | 74LS373 | 5 |
| U 7 | 74LS157 | 7 |
| U 8 | 74LS157 | 8 |
| U 9 | 74LS157 | 6 |
| U10 | 74LS165 | 9 |
| U11 | 74LS365 | 10 |
| U12 | 2332 | 2 |
| U13 | 74LS74 | 19 |
| U14 | 74LS93 | 21 |
| U15 | 74LS05 | 14 |
| U16 | 74LS05 | 15 |
| U18 | 74LS86 | 20 |
| U19 | 74LS04 | 13 |
| U20 | 74LS10 | 16 |
| U21 | 74LS00 | 11 |
| U22 | 74LS00 | 12 |
| U23 | 74LS32 | 17 |
| U24 | 74LS74 | 18 |
| U25 | 7805 | 22 |

MicroAce ICs 4, 5 & 17 are used for extra 1K RAM onboard.



## 4K/1K GRADEBOOK FOR TEACHERS

Just in time for fall--this 4K/1K program helps teachers average grades at the end of the grading period. It uses the high precision division routine in the 4K ZX80 manual.

Type it in, then run. Enter the gradebook page number in response to the first prompt, then the line number to identify each student. After you enter all the grades for any student, type any negative number (greater than -32768) as the last grade. The program jumps to the division routine and calculates the average to three decimal places. To go on to the next student, hit CONT (NL) CONT(NL).

Robert M. DeMunbrun, Rushville, IN

```
   1 PRINT "GRADEBOOK PAGE NUMBE
R IS ";
   2 INPUT PG
   3 PRINT PG
   4 PRINT "GRADEBOOK LINE NUMBE
R IS ";
   5 INPUT ID
   6 PRINT ID
  10 LET N=0
  20 LET G1=0
  25 PRINT "GRADE"
  30 INPUT G
  40 IF G<0 THEN GO TO 300
  50 PRINT G
  60 LET N=N+1
  70 LET G1=G+G1
  80 GO TO 30
 300 LET X=G1
 310 LET Y=N
 320 LET Z=X/Y
 330 LET R1=X-Z*Y
 340 LET D1=10*R1/Y
 350 LET R2=10*R1-D1*Y
 360 LET D2=10*R2/Y
 370 LET R3=10*R2-D2*Y
 380 LET D3=10*R3/Y
 390 PRINT "THE AVERAGE IS---";Z
;".";D1;D2;D3
 400 STOP
 410 GO TO 2
```

Syntactic Sum: 26666, 4K

## BINGO NUMBER GENERATOR—4K

Do you have an old bingo game in your closet with lost number tokens? Here is a program to render your old bingo cards useful again. It would be very helpful for community bingo games, allowing the hearing impaired or deaf to see the numbers on the video monitor.

It generates numbers with their preceding letters and prints them at 8x their normal size at the top of the screen. To get another number, just hit (NL). To see a list of previously generated numbers, press L (NL). Follow the prompts for other features. To exit, press (NL) then BREAK.

This program uses 4K ROM and 3K RAM. If you delete lines 612 and 695, it will run in 2K.

Richard Van Workum, Hanford, CA

```
  1 DIM C(3)
  5 DIM S(75)
 10 DIM A(75)
 15 FOR G=1 TO 75
 20 LET S(G)=0
 25 LET A(G)=0
 30 NEXT G
 35 CLS
 37 RANDOMISE
 40 LET H=RND(75)
 45 LET A(H)=H
 50 IF A(H)=S(H) THEN GO TO 40
 55 LET S(H)=H
 60 IF A(H)<16 THEN LET C(1)=39
 65 IF A(H)>15 AND A(H)<31 THEN
LET C(1)=46
 70 IF A(H)>30 AND A(H)<46 THEN
LET C(1)=51
 75 IF A(H)>45 AND A(H)<61 THEN
LET C(1)=44
 80 IF A(H)>60 THEN LET C(1)=52
 90 IF A(H)<10 THEN GO TO 115
100 LET C(2)=H/10+28
105 LET C(3)=H-(H/10)*10+28
107 LET N=3
110 GO TO 120
115 LET C(2)=H+28
116 LET N=2
120 FOR L=2 TO 7
130 FOR D=1 TO N
140 LET Q=PEEK(C(D)*8+3583+L)
150 LET B=10
160 LET B=B-1
170 IF B<0 THEN GO TO 240
180 IF Q<2**B THEN GO TO 220
190 LET Q=Q-2**B
200 PRINT CHR$(128);
210 GO TO 160
220 PRINT " ";
230 GO TO 160
240 NEXT D
245 PRINT
250 NEXT L
251 FOR E=1 TO 15
252 PRINT
253 NEXT E
255 PRINT "PRESS (NL) FOR NEXT
NUMBER"
256 PRINT "PRESS L, (NL) FOR LI
ST"
260 INPUT A$
270 IF A$="" THEN GO TO 35
275 IF NOT A$="L" THEN GO TO 26
0
280 CLS
303 LET W=0
304 LET X=0
305 LET Y=0
306 LET Z=0
307 PRINT "B    I    N    G
O"
308 PRINT "-- 3 -- 3 -- 3 -- 3
--"
310 FOR V=1 TO 15
320 IF A(V)=0 THEN GO TO 350
329 IF A(V)<10 THEN PRINT " ";
330 PRINT A(V);" 3 ";
340 IF W>0 THEN GO TO 410
345 GO TO 370
350 NEXT V
360 IF V>15 THEN PRINT "  5  ";
365 IF W>0 THEN GO TO 410
370 FOR W=16 TO 30
380 IF A(W)=0 THEN GO TO 410
390 PRINT A(W);" 3 ";
400 IF X>0 THEN GO TO 470
405 GO TO 430
410 NEXT W
420 IF W>30 THEN PRINT "     ";
425 IF X>0 THEN GO TO 470
430 FOR X=31 TO 45
440 IF A(X)=0 THEN GO TO 470
450 PRINT A(X);"   ";
460 IF Y>0 THEN GO TO 530
465 GO TO 490
```

```
 470 NEXT K
 480 IF X>45 THEN PRINT "      ";
 485 IF Y>0 THEN GO TO 530
 490 FOR Y=46 TO 60
 500 IF A(Y)=0 THEN GO TO 530
 510 PRINT A(Y);"   ";
 520 IF Z>0 THEN GO TO 585
 525 GO TO 550
 530 NEXT Y
 540 IF Y>60 THEN PRINT "      ";
 545 IF Z>0 THEN GO TO 585
 550 FOR Z=60 TO 75
 560 IF A(Z)=0 THEN GO TO 585
 570 PRINT A(Z)
 580 GO TO 350
 585 NEXT Z
 590 PRINT
 595 IF V>15 AND W>30 AND X>45 A
ND Y>60 AND Z>75 THEN GO TO 610
 600 GO TO 350
 610 LET F=-15
 612 LET P=0
 615 FOR M=1 TO 5
 620 LET F=F+15
 625 LET G=0
 630 FOR E=1+F TO 15+F
 635 IF NOT A(E)=0 THEN LET G=G+
1
 640 IF G>P THEN LET P=G
 645 NEXT E
 650 NEXT M
 685 FOR E=1 TO 18-P
 690 PRINT
 695 NEXT E
 697 PRINT "PRESS (NL) FOR NEXT
NUMBER"
 700 PRINT "PRESS G, (NL) FOR NE
W GAME"
 705 INPUT A$
 710 IF A$="" THEN GO TO 35
 715 IF A$="G" THEN RUN
 720 GO TO 705
Syntactic Sum: 10563, 4K
```

ANNOTATED 4K ROM LISTING—SYSTEMS VARIABLES

Next in our series of excerpts from Sinclair's 4K ROM listing is systems variables. This listing tells you the values of all major variables used in the 4K ROM. Note the designer assigned the value of Y (the first variable) as 4000h, then defined all following variables relative to Y. As usual, because our assembler does not permit underline characters, we substituted 8s. So E8LINE, for example, is really E_LINE and D8FILE is D_FILE.



```
                00100                           ;SYSTEM VARIABLES
4000            00110 Y         EQU     4000H   ;FIRST BYTE OF RAM & VALUE
                00120                           ;ALWAYS HELD IN IY
4000            00130 ERR8NR    EQU     Y       ;RUN TIME ERROR # - 1
4001            00140 FLAGS     EQU     ERR8NR+1        ;D0=SUPPRESS SPACE
                00150 ;BEFORE ALPHA TOKENS WHEN PRINTING
                00160 ;D2=KEYWORD CONTEXT AT CURSOR
                00170 ;D3=REQ JUMP TO STMT # IN PPC
                00180 ;D5=IDENT FOUND AFN CALL RATHER THAN A VARIABLE
                00190 ;D6=TYPE OF (RESULT) IS INTEGER RATHER THAN STRING
                00200 ;D7=REQ INTERPRET RATHER THAN CHECK SYNTAX
4002            00210 PPC       EQU     FLAGS+1 ;LINE # OF INSTRUCTION BEING
                00220                           ;INTERPRETED
4004            00230 P8PTR     EQU     PPC+2   ;CURSOR IN LINE BEING EDITED
4006            00240 E8PPC     EQU     P8PTR+2 ;LINE # OF CURRENT LINE IN
                00250                           ;LISTING
4008            00260 VARS      EQU     E8PPC+2 ;FIRST RECORD FOR A VARIABLE
```

```
                00270                           ;(LAST IS BYTE 80H)
400A            00280 E8LINE    EQU     VARS+2  ;LINE BEING EDITED
400C            00290 D8FILE    EQU     E8LINE+2        ;NL THAT PRECEDES 1ST
                00300                           ;LINE OF DISPL FILE
400E            00310 DF8EA     EQU     D8FILE+2        ;FIRST CHAR IN 2ND
                00320                   ;HALF OF SCREEN EDIT AREA
4010            00330 DF8END    EQU     DF8EA+2 ;BYTE AFTER DISPLAY FILE
4012            00340 DF8SZ     EQU     DF8END+2        ; # LINE IN 2ND HALF
                00350                   ;OF SCREEN INCL SEPARATING BLANK LINE
4013            00360 S8TOP     EQU     DF8SZ+1 ;LINE # (IN PROGRAM) OF TOP
                00370                   ;LINE ON SCREEN
4015            00380 X8PTR     EQU     S8TOP+2 ;FIRST CHAR NOT SYNTACTICALLY
                00390                   ;OK (0 IF ALL OK)
4017            00400 OLDPPC    EQU     X8PTR+2 ;LINE # OF E.G. INTERRUPTED
                00410                   ;STATEMENT
4019            00420 FLAGX     EQU     OLDPPC+2        ;D5=REQ INPUT VALUE
                00430                   ;NOT LINE OF PROGRAM
                00440                   ;D6=REQD TYPE
                00450                   ;D7=KEYWORD CONTEXT AT (CH8ADD)
401A            00460 T8ADDR    EQU     FLAGX+1 ;NEXT BYTE IN TEMPLATE
401C            00470 SEED      EQU     T8ADDR+2        ;LAST RANDOM # BEFORE
                00480                   ;SCALING
401E            00490 FRAMES    EQU     SEED+2  ;INCREMENTED ONCE PER FRAME
                00500                   ;WHILE DISPLAYING
4020            00510 DEST      EQU     FRAMES+2        ;FIRST CHAR OF
                00520           ;VARIABLE MATCHED BY TEMP CODE 1 OR 4
4022            00530 RESULT    EQU     DEST+2  ;VALUE YIELDED BY EPTRN OR
                00540                   ;FIND SCREEN POSITION
4024            00550 S8POSN    EQU     RESULT+2        ;(LINE & COL) OF NEXT
                00560                   ;CHAR TO BE OUTPUT
4026            00570 CH8ADD    EQU     S8POSN+2        ;CURRENT CHAR WHEN
                00580                   ;SYNTAX CHECKING ETC
4028            00590 RAMBOT    EQU     CH8ADD+2        ;FIRST BYTE OF USER'S
                00600                   ;RAM AREA--1ST BYTE OF PROGRAM
                00610                   ;OTHER EQUATES
0020            00620 L8LEN     EQU     32      ; # CHARS PER LINE ON DISPLAY
001C            00630 DIGIT     EQU     1CH     ;DIGIT+N IS CODE FOR DIGIT N
0025            00640 LETTER    EQU     25H     ;LETTER+N IS CODE FOR NTH
                00650                   ;LETTER OF THE ALPHABET
0076            00660 NL        EQU     76H     ;CODE FOR END-OF-LINE ON
                00670                   ;DISPLAY AND IN PROGRAM TEXT
00E6            00680 MIN8KW    EQU     0E6H    ;FIRST TOKEN THAT IS A KEYWORD
                00690                   ;RATHER THAN AN OPERATOR
00B0            00700 CURSOR    EQU     0B0H    ;CURSOR IN LINE BEING EDITED
                00710                   ;(A DD.1 FOR LETTERS MODE)
0001            00720 QUOTE     EQU     1       ;STRING QUOTE
000D            00730 DOLLAR    EQU     0DH     ;DOLLAR SIGN
00D8            00740 COMMA     EQU     0D8H    ;TOKEN FOR ','
00D9            00750 KET       EQU     0D9H    ;TOKEN FOR ')'
00DA            00760 BRA       EQU     0DAH    ;TOKEN FOR '('
                00770 ;NB EXPRN ASSUMES '(','NOT','-' ARE ADJACENT
00DC            00780 MINUS     EQU     0DCH    ;TOKEN FOR '-'
00DC            00790 LO8OPR    EQU     MINUS   ;FIRST DIADIC OPERATOR
00E3            00800 EQUAL     EQU     0E3H    ;TOKEN FOR '='
00E5            00810 HI8OPR    EQU     0E5H    ;LAST DIADIC OPERATOR
```

DEAR EDITOR:

As a new subscriber, I wonder what I have missed in back issues. Do you have an index? Specifically, I need info on A/D-D/A. The ZX80's price is so low that I may be able to use it for control functions. Can you help?

Neal Immega, Houston, TX

You'll get an index to Vols. 1 and 2 with the Dec.81 issue. After that, we'll publish an annual year-end index to all programs and articles in SYNTAX. We have not published information about A/D-D/A in SYNTAX. For low precision applications, companies such as Analog Devices, Wilmington, MA, and Datel, Canton, MA, make subsystem chips designed to connect to 8-bit micros. Some electronic supply houses also sell DVM chips that you can interface to your computer as an A/D device. If you can live with the fixed voltage steps, try Radio Shack's LED bar drivers.—KO

---

I give up. What is the purpose of the 1K resistor R24 in my Sinclair ZX80?

Dean A. Cuadra, Los Angeles, CA

Sinclair's technical information people told me it just reduces the current from the NOT gate to the AND gate.—AZ

---

Please warn your readers about the pitfalls of ordering merchandise from some English firms. Case in point:

On May 7, 1981, I mailed 3 international money orders, one to Phipps Associates for the ZX80 Pocketbook, one to Timedata Ltd. for the ZX80 Magicbook, and one to Zipprint for the ZX80 Programs Vol 1. In 3 weeks I received the book from Phipps with no problem. But the other 2 firms failed to send me the books ordered or any message advising of a shipping delay.

On July 10 the bank told me the money orders for Timedata and Zipprint had been paid. When nothing arrived by August 9, I wrote to each firm, telling them that they had cashed my check and I was still waiting for my books. As of September 9, 1981, I have received neither the 2 books nor the courtesy of a reply.

Also, I wrote to request catalogs and information from Science of Cambridge and Linsac and received no reply after 4 months.

To be fair, I have done business with the following English firms and have been completely satisfied: Macronics, Datalog, National ZX80/1 Users' Group, and Phipps Associates.

It's a shame that we order merchandise in good faith, with the money up front, only to find that some firms are less than honest.

John A. Sampson, College Point, NY

---

I wonder if your readers have tried running R. Bissell and K. MacDonald's Amazing Active Display program on p.68 of Tim Hartnell's Making the Most of Your ZX80? I am unable to run it successfully on my 2K MicroAce. Using the parameters suggested for A, C and T results in a brief flicker then a system crash, surely not the desired effect! Pressing BREAK doesn't recover the lost program. Increasing A for larger RAMs as suggested provides results varying from rolling horizontal bars to recovery of a garbled form of the BASIC program. Any suggestions?

W.A. MacDougall, Ottawa, Ontario

Carl Rasmussen of DePere, WI, also wrote to say he cannot run Amazing Active Display. Our copy of Hartnell's book has not arrived; we'll let you know what happens when we run that program. Until then, has anyone run it successfully?—AZ

Some users have interference problems with the TV display. The fine tuning won't help. Simply connect a ground strap from the transfer switch box (game-TV box) to the tuner body. Now the display is steady and clean.

However, I do have loading problems. No program, my own or Sinclair's, will load from tape.

What is required to upgrade the system to 16K RAM?

Tony Wendels, Van Nuys, CA

We don't recommend your solution to interference problems. It can directly connect the transfer switch case to the AC line, possibly causing shock.

All you need to upgrade your ZX80 to 16K RAM is $99.95. The RAM pack comes with the extra power supply it needs, and will work with either 4K or 8K ROM. It just slides on the edge connector at the left rear of your computer.—AZ

---

I discovered a problem with my MicroAce that may also affect ZX80s. When I hooked up the 8K memory I built, the characters would blink on and off or I would get nothing on the screen.

The problem was a bad connection between the power plug and the power jack on the computer. My solution was to cut the plug off, remove the jack and solder the wires to the PC board. The positive lead has a white stripe on it, and the PC board is marked "+" and "-" on the bottom side.

After I did this, the screen cleared up better than it has ever been, even with the 8K memory hooked up. This idea might clear up a lot of elusive problems.

Jimmy L. Droit, Marion, IL

---

I have run into a problem with my ZX80 I haven't seen discussed anywhere. After I type perhaps 20 or 30 program lines, it starts erasing a character every time I enter another one. When I press SHIFT RUBOUT it adds back the character it just erased up to the point at which the malfunction began, then it will begin erasing. I assumed it might have been from of lack of memory, however, it happens after I plug in the 16K RAM. I have certainly not exceeded the memory capacity of that unit.

I also noticed another strange thing. The K cursor jumps from the bottom of the screen to the middle or upper third after entering the second or third line number. This did not seem to affect the function until I reached the point where the enter and rubout problem began.

Brooks A. Mick, Findlay, OH

Both problems sound like too little memory. The K cursor will move up and jump around the screen as memory dwindles yet it will continue to function. When the ZX80 runs out of memory, the characters you type will not appear on the screen. You shouldn't run out of memory with the 16K RAM pack; try running the memory connected program Sinclair sent you with the unit to be sure the connection between the RAM and the ZX80 is good. If it isn't, try our suggestions on p.6 of the Sept.81 issue.—AZ

---

I purchased a used ZX80 from Sinclair as advertised in SYNTAX (Aug.81) and I need some replacement parts to get it working. I got exact replacement of the voltage regulator IC22 from Radio Shack (#276-1770). But I need to know where to get other parts, particularly the X1 crystal, 6.5 MHz.

H.S. Wake, San Diego, CA

Sinclair stocks some parts as replacements for their kits. Contact them at 50 Staniford St., Boston, MA, 02114, or call 617/742-4826.—AZ

These SYNTAX readers would like to contact others. If you'd like to hear from users in your area, send us your name, address, phone number and ROM/RAM size. We'll publish them when space permits.—AZ

*Rich Reinsch, 1624 G St., Geneva, NE, 68361, is interested in 8K/16K programs.

*Steve Matta, P.O. Box 4644, Shreveport, LA, 71104-0644, 318/869-5587.

## 8K ROM MONITOR ROUTINES

(Part 2 of an article on 8K routines. See Part 1, Sept.81--AZ)

'Expand Display File': The ZX81 (or 8K ROM) collapses its display file when times are hard--not enough memory. With a machine with less than 3.25K of RAM (genuinely or because you moved RAMTOP down) the display file is kept at only 25 NEWLINE characters. With a machine with more memory the display file is fully expanded (most of the time).

Because the display is or may be collapsed, the 8K monitor program must repeatedly test for the existence of the print position. If it is not there, the ROM must fill out the line so that the print position comes into being.

The RST 0010 routine (Sept.81) does this for every character that passes, but it does so by simply incrementing its existing values before calling the "expand if necessary" routine. However, the PRINT AT command allows you to choose any place in the display for the next print position.

This program shows how the PRINT AT operation can be performed. (In line 10, the first symbol is shift 1, the second shift D. Use the functions LN, NOT & TAN; don't type them out.)

```
  10 REM ▪5/LN /▬Y*NOT TAN
  20 LET L=USR 16514
```

This places a * in the upper left position. The machine code in line 10 is:

```
   LD   BC,2118 ;PRINT AT 33,24
   CALL 0918    ;inverted
   RST  0010
   RET
```

Note that at this entry point the coordinates are reversed.

Get the PRINT AT positions oriented the proper way, then use the entry point of the "TEST PRINT AT values" routine at 08F5h. These lines will PRINT AT 12,12 (the first symbol is shift 1, the second shift A. Use the function key for LN, PRINT, NOT & TAN; don't type them out).

```
  10 REM ▪ ££LN PRINT ▩Y*NOT TAN
  20 LET L=USR 16514
```

To type line 10, start with 10 PRINT. Backspace and add REM.

'Print String': The 8K ROM PRINT command routine at 0ACFh scans the PRINT statement and prints the required characters. All printing uses the RST 0010 instructions (Sept.81).

When you want to print a string, the DE register pair must hold the starting address of the text of the string. The BC register pair must hold the number of characters in the string. Then call 0B6Bh.

Try this program to see how 0B6B prints strings:

```
  10 REM SYNTAX ▪▛ )▙RNDLN ?"TAN
  20 POKE 16528,107
  30 LET L=USR 16520
  40 GOTO 30
```

(The symbols are shift 1, shift E, and shift W. Use the function key for RND, LN and TAN.)

Here's the machine code translation for line 10:

```
  LD   BC,+0007  ;7 string char.
  LD   DE,+4082  ;address of the S
  CALL 0B6B      ;print the 7 char.
  RET            ;return to BASIC
```

The instruction 6B (decimal 107) cannot be entered from the keyboard, so you must POKE it into its address. Of course, you could enter all the instructions using

POKE--it's a matter of choice.

Try another example:

```
10 REM ■ FOR  ))■ LN ?"TAN
20 POKE 16521, 107
30 LET L=USR 16514
```

Again, for line 10 enter FOR, then backspace and enter REM. This program prints the command table as a string.

The identical BASIC program, which runs more slowly, is:

```
10 FOR A=273 TO 507
20 PRINT CHR$ PEEK A;
30 NEXT A
```

Note the last letter of each command word is in reverse video. The monitor uses this technique to show the end of the word.

Ian Logan, Skellingthorpe, UK

(Dr. Logan added that he is interested in getting letters about any of his articles, and will try to answer all letters. His address: 24 Nurses Lane, Skellingthorpe, Lincoln, LN6 0TT, UK.)

SOFTWARE REVIEW

A Night in Las Vegas

```
            Price:  $9.95
         RAM reqd:  1K
         ROM reqd:  4K
  Type of program:  Games
Printed listings?:  No
Program listable?:  One
  Screen prompts?:  Yes
        Challenge:  High
          Display:  Good
```

From: Lamo-Lem Laboratories, Box 2382, LaJolla, CA. 92038.

Lamo-Lem's Night in Las Vegas is definitely a cut above the usual 4K ROM game. The package includes the casino standards, Slot Machine, Craps, Blackjack and Roulette. You get three copies of each game on a single cassette, although I had no trouble loading any of them. You also get a complete book of instructions, explaining how to load and operate the programs, and also how to play the games (handy for your next trip to Atlantic City). A set of chips is included for playing roulette on the large roulette table board and craps table sheet provided.

Slot Machine is just like the real thing--you pump money into the one-armed bandit. The program nicely displays a slot machine on the screen and the fruits (graphic symbols) come up one at a time.

Blackjack offers more challenge (and chance to win), since you play against the machine. The display is not quite as good, and you will be dealt ones. But Lamo-Lem got four decks and a re-shuffle feature into 1K RAM--an admirable feat. And the machine extends you $100 easy credit if you go broke.

Roulette lets you place up to five bets, keeping track of your total money after each. You get all the betting options of a roulette table. After the last bet, the wheel spins--and you see the numbers displayed on the screen as the ball goes by them, slowing down until the winner comes up.

You then see your bets and their payoffs, and your total bankroll.

Craps comes with what seems to be a thorough set of instructions, but despite my misspent childhood, I still don't understand how to play the game. The program allows you to bet the pass line, take the odds, bet any craps, or the big 6 or big 8, among other choices. It displays the results of both dice and your remaining money.

Lamo-Lem again provides excellent keyboard overlays, this time in felt green. By placing them over the computer's keyboard, you are sitting at a blackjack, roulette or craps table--a small touch that really improves the play.

For $9.95, this is an inexpensive way to indulge your gambling vice or just have fun.—AZ

BOOK REVIEW

Title: Crash Course in Microcomputers
By: Louis E. Frenzel, Jr.
From: Howard W. Sams & Co., Inc.
Price: $17.50 (paper/spiral-bound)

Before we start, let me tell you up front that SYNTAX will be selling this book. In my view, we're selling the book because it's good, not because it's good to sell. So, knowing my potential for bias, here's the review.

Sams' Crash Course in Microcomputers is for users of widely varying backgrounds. As author Frenzel explains, "As for prerequisites, there are none. No previous knowledge of computers or electronics is required to use this book. Yet it is written at a level

that is acceptable to average consumers as well as scientists holding PhDs."

This course deals with all of microcomputing, not just programming. It treats hardware and machine language first, BASIC and applications last. The programmed learning format and self-tests make it easy to learn about micros.

Hardware described in this book includes the Z80 processor used in your ZX80. Machine language instruction uses the 8080 subset of Z80 commands. Thus you can directly apply these sections to your machine. In addition, the book covers input/output operations and devices--such things as ports and memory-mapped I/O.

Frenzel chose Tiny BASIC as the high-level programming language, fitting nicely with your machine. Tiny BASIC has only one feature (INPUT A,B,C) not supported by Sinclair, so translating programs from Tiny BASIC to ZX80 BASIC is possible with almost no change. If you add changes using the ZX80's more powerful BASIC, programs will generally get shorter.

Frenzel assumes you have floating point arithmetic, so use with the 8K ROM is more direct than with the 4K integer BASIC. No matter--the author expects you to execute these simple programs with pencil and paper anyway. And you are only brought to the level of writing a Fahrenheit to Celsius conversion program.

You'll find the four-page applications unit very short, sufficient only for an overview.

Chapters run 12-26 pages, including a self-test and answers. Appendices describe the 8080 instruction set and ASCII code.

When you finish this book, you should be able to define, describe and understand all of the terms common to microcomputer systems. Your ability to read and understand ads and articles will be significantly enhanced.—KO

## BEGINNERS: USER-FRIENDLY PROGRAMS

Experienced programmers know that the best programs are not necessarily those with the most clever structure. Good programs do the required job and make it easy for the user. Programs that are simple to run and understand the results of are "user-friendly." This month we'll look at some easy programming methods to write user-friendly programs for either 4K or 8K ROM programs.

Almost all programs require some input from the user to run. You may have to enter numbers or other information. When you run a program on the ZX80, the cursor indicates when the machine expects input--LS for numbers (or numerical variable) and "L" for strings (usually letters) on 4K machines, L for numbers and "L" for strings on 8K machines. A good program gives prompts for the user; it tells you what information it needs.

Prompts are simply PRINT statements asking for information. Just use a line like:

```
10 PRINT "WHAT IS YOUR NAME?"
```

The next line tells the computer to accept input from the keyboard:

```
20 INPUT A$
```

INPUT A$ means that the user will enter a literal string, so each time it prints A$, it will print just what you type in. String variables consist of any single letter, A-Z, followed by a dollar sign.

Now add:

```
30 PRINT "HELLO, ";A$
```

and the computer calls the user by name--how much friendlier can you get?

This method also works for entering numbers, except that the input is not a string but a numerical variable:

```
10 PRINT "ENTER AMOUNT"
20 INPUT A
```

Numerical variables can be any length, composed of letters A-Z and digits 0-9.

After inputting a number, another nice touch is to print the number for the user to see.

```
  30 PRINT A
```

or:

```
  30 PRINT "AMOUNT=";A
```

To remove the prompt, insert:

```
  25 CLS
```

before the second print statement so the user sees only his response displayed on the screen.

For users not experienced in handling computers, programs that allow you to easily exit or rerun a program are helpful. Add these lines to the end of your program, using appropriate line numbers. At the end of a game, for example, add:

```
 500 PRINT "PLAY AGAIN? Y OR N"
```

This line gives the user the option of running the program again and tells him how to respond. Again, follow this with an INPUT line (a string variable because the answer will be a letter):

```
 510 INPUT Z$
```

Now you tell the computer how to interpret the user's response:

```
 520 IF Z$="Y" THEN GO TO (first
line number)
 530 IF Z$="N" THEN STOP
```

Another way to run the program again is to use the RUN command:

```
 520 IF A$="Y" THEN RUN
```

This method clears any variables the user entered during the first run. It's just like typing RUN from the edit mode (when the listing is on the screen). It's a good idea to put a CLS at the front of the program to remove any left-over display from the last run.

Another elegant touch to a BASIC program is adding tests for valid inputs. For example, if the program asks for a number between 1 and 10, you don't want it to continue if the user enters 500. Here's one way to test inputs.

First, of course, insert a prompt to ask for the desired input and the input command for the computer:

```
  10 PRINT "CHOOSE A NUMBER BETW
EEN 1 AND 10"
  20 INPUT X
```

Now for the test:

```
  30 IF NOT X*2<21 AND X*2>0 THE
N GO TO 10
```

Any number between 1 and 10 multiplied by 2 will be less than 21 (the largest acceptable number, 10, times 2 is 20.) And any number greater than 1 is positive, so multiplying it by 2 will be greater than 0. If both of these are true, the number must be 1 or 10 or a number between the two. If both are not true, the computer goes back to line 10 and again asks the user to input a number.

To test a string variable for validity, use the same method:

```
 500 PRINT "PLAY AGAIN? Y OR N"
 510 INPUT Z$
 520 IF NOT Z$="Y" OR Z$="N" THE
N GO TO 500
```

Line 520 tests to see if the user's response is either a Y or an N. This time Z$ can be either of two choices, so you use OR. Since the response is not valid only if it isn't a Y or an N, you use IF NOT. If the response isn't valid, the computer goes back to line 500, which asks the question again until it gets an answer it expects.

These methods are not the only ways to accomplish their goals, but they do work. If you have enough RAM to fit them in your program, using them will make it easier for others to use your program.

## SINCLAIR PRINTER BY 1982

Sinclair's ZX81 printer is in production in the UK, but we won't see it in the US until the end of 1981. The printer has not yet been submitted to the FCC for approval. Sinclair engineers will redesign the printer to meet US requirements at the end of September, 1981.

---

Now that 8K ROMs and ZX81s will be widely available, we're looking for more 8K programs for SYNTAX. Send your 8K programs for 1-16K RAM!

## 8K/1K SORT ROUTINE

This 8K/1K routine can sort 20 words of 20 characters in about 8 seconds. This program was originally an Apple sort program but required major revisions to run on the ZX80.

If you use it to order numbers, make sure they have the same number of digits, filling in leading zeros if necessary. Spaces between letters will be considered in the sort. The ZX80 gives numerical values to letters and numbers. For example, A is less than B. A space is less than any letter.

Line 200 allows the program to be saved on tape. Just type RUN 200 to save. To load, type LOAD "SORT". The program will come up ready to run. To exit at the end of program, type (NL). To exit during a sort, type BREAK (NL).

Leo Morgan, Peabody, MA

```
   1 REM ALPHA-NUMERIC SORT ROUT
INE REVISED BY LEO MORGAN 9/5/81
   5 PRINT "ENTER NUMBER OF WORD
S, 1 TO N"
  10 INPUT N
  15 DIM A$(N,20)
  20 FOR I=1 TO N
  25 PRINT "TYPE A WORD, 1 TO 20
 CHARACTERS"
  27 PRINT AT 21,0;I
  30 INPUT A$(I)
  35 CLS
  40 NEXT I
  50 LET X=0
  60 LET X=X+1
  70 IF X>N THEN STOP
  80 IF A$(X)="ZZZZZZZZZZZZZZZZ
ZZZ" THEN GOTO 60
  90 FOR Y=1 TO N
 100 IF A$(Y)<A$(X) THEN LET X=Y
 110 NEXT Y
 120 PRINT A$(X)
 130 LET A$(X)="ZZZZZZZZZZZZZZZ
ZZZZ"
 140 GOTO 50
 200 SAVE "SORT"
 210 GOTO 5
Syntactic Sum:  24438, 8K
```

## A TOOL TO HELP DEBUG PROGRAMS

To determine what occurs during a loop portion of a malfuntioning program, use a PRINT statement called a flag.

Suppose the 8K Sort routine does not work properly--line 70 won't stop the program. To find out what's happening, type:

```
  72 PRINT " "," ","72X=";X
```

The computer will tell you the value of X when it gets to line 70. This way you can track down what went wrong.

Or perhaps line 110 is a problem--type:

```
 112 PRINT " "," ","112Y=";Y
```

to see the value of Y. Put your flags after problem lines involving variables to check their status.

Leo Morgan, Peabody, MA

## ROUND OFF ROUTINE—8K

The 8K ROM's floating point decimal is great, but sometimes you don't need 9 decimal places. Use this routine in programs involving dollar amounts to round to 2 decimal places. It also pads empty decimal places with zeros and lines up decimal points.

To adjust decimal places to, for example, three places, change 100 to 1000 in lines 20 and 30, change 2 to 3 and 0 to 00 in line 50, and increase the constant subtracted from L in line 70 by one.

Jon Passler, Beverly, MA

```
  10 INPUT N
  20 LET N1=INT (N*100)
  30 IF N*100-N1>=.5 THEN LET N1
=N1+1
  40 LET N$=STR$ N1
  50 IF LEN N$<2 THEN LET N$="0"
+N$
  60 LET L=LEN N$
  70 PRINT TAB 10-L;N$( TO L-2);
".";N$(L-1 TO )
  80 GOTO 10
Syntactic Sum:  11646, 8K
```

## CLASSIFIED ADS

SYNTAX CLASSIFIEDS: Just $2.75/line (4 line min), send ad, typed 35 char/line, with payment by 15th of month for next month's issue. SYNTAX, RD 2 Box 457 Bolton Road, Harvard, MA, 01451.

Programs-games & utility. Also tech data, mods, plans, info,etc. Send SASE for free goodies list P.O. Box 3073 San Jose, CA 95156

MX-16, 16K RAM expansion for the ZX-80. To order send check or money order to INSIGHT, 1889 Lewis Dr., Niles, MI 49120. For C.O.D. call 616-674-7868. Just $89.95 plus $3.00 shipping and handling. No extra charge for C.O.D.

For Sale: ZX-80 (4KROM &1KRAM), subscription to SYNC + back issues of SYNTAX to 9/81. Like NEW $175 Call Rich 617-359-4429 nights

Fascinating LIFE graphics program, 22x27 world, half second between displays. $1 for listing + instructions + patterns-4K/1K. Jon Passler 344 Cabot St., Beverly, MA 01915.

ZX80/81 "Record"--at last a tape record system! Save, load or enter new 96 byte records. Ideal for addresses, etc. Listing for all 1K machines (4K/8K ROM) $7. "Directory"--a simple program to read tapes and display program names. Listing (8K ROM) $5. Logan Software, 24 Nurses Ln, Skellingthorpe, Lincoln, LN6 0TT, UK

SALE: ZX80 4K ROM & Wired Keybd. ZX80 Comp. book & 10 issues of SYNTAX & 4 SYNC, Shubert Kersey 904 4th Ave, McRae, GA 31055 All $ 140 + shipping

7 QUALITY GAMES on CASSETTE--$8.95 Mastermind (Play any difficulty up to 6 out of 9), Doublemind, Slot Machine, Craps, Tic Tac Toe (Challenge the computer), Sub Rescue, White Hot Number (Pit your score against the computer's). All run in 4K ROM/1K RAM. NEW ENGLAND SOFTWARE, Box 691, Hyannis, MA 02601.

From SYNTAX: Zilog Z80-Z80A CPU Technical Manual, $7.50, Z80-Assembly Language Progamming Manual, $15. Add 5% for postage and handling. Sinclair's 4K ROM listing with designer's comments, $40, Crash Course in Microcomputers, $17.50 plus $1.50 shipping. SYNTAX Vol 1 (Nov/Dec.80) $5. Other back issues, $4 each. Send check or credit card no. to SYNTAX, RD 2 Box 457 Bolton Rd., Harvard, MA.



SYNTAX

THE HARVARD GROUP
Bolton Road, Harvard, Mass. 01451

**First Class**